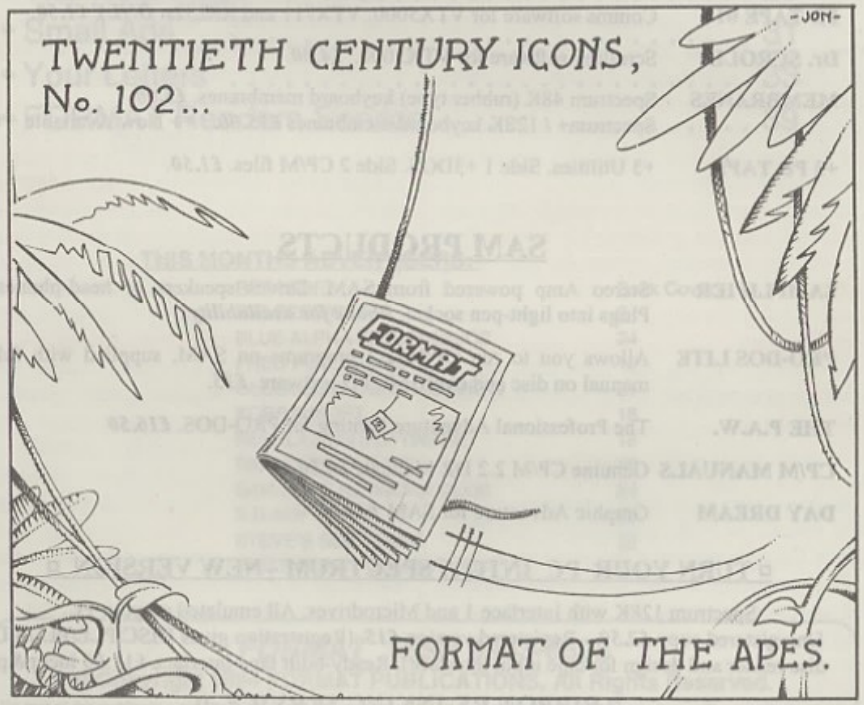
Vol.7 №8. April 1994.

FORMAT

FOR SPECTRUM AND SAM USERS



Get Into The Swing
With ***FORMAT***

£1.25 (UK R.R.P.) ISSN 0963-8598.

CONTENTS

Vol.7 N°8. **April 1994.**

- News On 4 4
- The Editor Speaks 5
- Short Spot 7
- Machine Code Without The Tears - Part 25 . . . 13
- BASICally Speaking - Part 6 19
- UniDos Corner 27
- The Help Page 29
- Small Ads 31
- Your Letters 33
- FORMAT Readers Service 39

THIS MONTHS ADVERTISERS:-

BETASOFT	Back Cover
B.G.SERVICES	2
BLUE ALPHA ELECTRONICS	24
FRED PUBLISHING	12
GLOUCESTER'S APRIL SHOW	21
KOBRAHSOFT	16
REVELATION SOFTWARE	18
RICHARDSON & CO	16
SAM SPEC NETWORK CLUB	24
S.D.SOFTWARE	26
STEVE'S SOFTWARE	32
WEST COAST COMPUTERS	6

FORMAT **ISSN 0963-8598**

FORMAT **is published by FORMAT PUBLICATIONS.**
34, Bourton Road, Gloucester, GL4 0LE, England.
Telephone 0452-412572. Fax 0452-380890.

Printed by D.S.Litho. Gloucester. Telephone 0452-523198.

NEWS ON 4

GLOUCESTER IN THE HEADLINES

As most of you will know, Gloucester has been hitting the news headlines over the last month - for all the wrong reasons. The 25 Cromwell Street murders have both horrified and transfixed the city and, so it seems, the rest of the world.

Gloucester, a normally quiet city, boasts one of the smallest police forces per head of population in the UK. Now the deeds of Frederick West have placed Gloucester in the spotlight, the media are having a field day and the police are under a lot of pressure.

Even though it is no longer front page material and the national TV news often has more pressing items to fill their bulletins with, the depressing thing is that, for Gloucester, the story goes on - day after day.

I'm sure readers will join with me in sending sympathies to the families of the poor victims.

ALL MOVE ROUND

It has been a case of musical chairs this month as both Blue Alpha and West Coast Computers start a big move.

Poor security at the Abernant site has led to major worries for both companies and so, after months of negotiations, it was decided to move the SAM production and development side, under Blue Alpha, to different premises. This leaves Abernant to be used for the storage of less nickable items by West Coast.

Blue Alpha Electronics address is now Ynysforgan Farm, Morriston, Swansea, SA6 6QL. A new telephone number will be available in the next week or so, ring us at ***FORMAT*** if you need help.

West Coast will continue to use Format as agents to handle the mail-order and dispatch side of their operation and our provision of office services for them is set to expand over the next six months as they increase their overseas sales operation.

At the same time West Coast have announced the launch of a new advertising campaign which will target console users, encouraging them to upgrade to a real Computer.

RED HERRING

This is an Adventure lovers dream - a thick bi-monthly magazine devoted to your favourite pastime. Red Herring is already on issue 15 and 16 is in the pipeline. The issue we obtained, number 14, ran to 88 professionally printed pages packed with every type of article an Adventure player could wish for. Certainly one of the best produced magazines I've seen in a long time.

The mag is not machine specific but many of the hints and ideas will serve you well no matter what machine you have.

Red Herring costs £4 per issue although if you subscribe for six issues (1 year) you get a seventh issue free. To order a sample copy send a cheque for £4, made payable to Marion Taylor, to Red Herring, 504 Ben Jonson House, Barbican, London, EC2Y 8NH.

Credits:-

URGENT we need your news. Anything you think other people should know about. Each item printed earns the contributor 3 months extra subscription (*please claim when next renewing*).

The Editor Speaks

As I write this life has just begun. Yes, the big FOUR OH. No big deal really, I don't feel any older than yesterday - I feel 75+ every day. And, it is only 25 years before I get my bus pass.

Since the demise of the High Street glossy mags there has been a glut of people starting 'so called' PD libraries for the Spectrum. One or two were run quite respectably but they, alas, did not survive very long (RAS PD and the like).

However, there seems to be rather a lot of kids (and yes I know one of the people I am talking about is well past his teens but from the way he is acting he deserves the title of 'kid') who are just getting PD a bad name. One recent case has nearly half a Spectrum disc mag devoted to slagging off the opposition.

A large file of letters is building up here, originating from one side or the other. Many demanding that I take action against the other side.

SO. To set the record straight. Unless I consider an advert to be either illegal, dishonest or misleading, or unless I receive specific complaints from readers about an advertiser, I have to run the adverts for anyone who is prepared to pay the list price. However, I do not have to give them any other form of publicity within these pages unless I approve of them. At the moment I feel quite strongly that PD on the Spectrum should not be encouraged. If software is worth having it should be published in a professional way so that money can be made to write new software. I would welcome readers comments on any PD company they have dealt with on the Spectrum. Give as much detail as you can. If you don't want your comments to appear in print then just say so, but it would help me to hear your views both bad and good (well there must be one good one out there somewhere).

On SAM the PD scene has been quiet for a long time. But I'm pleased to say that Derek Morgan is now well underway with his new SAM PD set-up (see letters page) and I hope that everyone will give him their support.

Many of you will remember the old days when Software for the Spectrum was available in almost every computer shop and you could buy a program to do almost anything you liked. Well, sad though it may be, those days are gone. Spectrum software is now mail-order only and that, strange though it may seem, is not a bad thing as it allows software to be cheaper as there are no middle men involved. One problem exists though, what happened to all the software that is no longer produced? In an attempt to answer that I would like to hear from readers who have details of unusual programs, we can then try to track down the author and make the software available again.

And finally. There has been a lot of problems this month due to the big move-around in South Wales (see news page) and it may still be a few weeks before everything is back to normal - in the wilds of South Wales BT take their time to connect new phone systems. Still, we are here - any problems we can help with we will. Just give us a ring.

Anyway. Show time is getting near. For more details see the advert. Be there or you will really miss out.

Until next month.

Bob Brenchley, Editor.

WEST COAST COMPUTERS

SPECIAL ANNOUNCEMENT

NEW SAM DISC DRIVES

You have all heard of the problems the slim-line drive shortage has caused SAM over the last two years -well we have the solution. Our design team has perfected a way of fitting standard 3½ inch disc drives economically (and cosmetically pleasing) to SAM. All new machines shipped since the beginning of March have been fitted with the new drives. A single drive 512K SAM, order code W1001, still costs ***£199.95*** but a dual drive SAM, order code W1002, has now come down in price to ***£259.95*** (both prices plus P&P).

UPGRADING TO TWO DRIVES

Because the fitting of these drives requires alterations to SAM's main circuit board, it is not now possible for users to fit second drives to their machines themselves. Machines will need to be sent to our repair agents Blue Alpha for upgrade. However we are keeping the price of a second drive at ***£79.95*** but this now includes fitting and return (insured) postage. Just give our mail-order agents at Format Publications a ring on 0452-412572 and they will tell you how to return your machine. Special arrangements are being made for overseas customers - please write for details.

NEW DRIVES FOR OLD

In tests, these drives are proving far more reliable then the old slim-line drives. We are therefore prepared to make you some special offers if you want to upgrade your existing drive at the same time as you have a second drive fitted.

If your SAM was purchased from us within the last six months. We will upgrade your existing drive for just £20 when you have a second drive fitted.

If your SAM is over 6 months old but still under our 12 month guarantee. We will upgrade your existing drive for a payment of £34.95.

If your SAM is over 12 months old. Providing the existing drive is working when we test it we will upgrade you for £49.95. Even if your existing drive is not working we will still offer you a £20 trade-in on the faulty one when you have two new drives fitted.

EXTENDED WARRANTY

While having new drives fitted, your SAM will be fully tested, if there are any faults discovered you will be notified and given a chance to have them corrected while we have your computer. Blue Alpha have also agreed that for a single payment of £10 they will give your machine a full 'parts and labour' warranty for 12 months. This only excludes the drives (which are covered by our 12 month guarantee) and the power supply/modulator, this will be in addition to any WCC warranty still left on your machine. Not a bad offer I think you will agree. Make your cheque for extended warranty payable to Blue Alpha Electronics.

For other items - Please see last months advert.

To order machines or any other items please send your order to:-
West Coast Orders, Format Publications, 34 Bourton Road, Gloucester, GL4 0LE.

SHORT • SPOT

YOUR HINTS, TIPS AND PROGRAMMING IDEAS

Edited By:- John Wase.

This month, I've lots and lots of lovely bits and pieces, but must first start with an apology. You know how it is when you are really really busy. Half a dozen extra jobs immediately arrive. So it was when Bob rang and mentioned that he wanted March 'Short Spot', pronto. I was chuffed to death; did it in record time, packed it up, a disc with all the SAM stuff on and a disc with all the PLUS D stuff - Bob likes 5¼" discs for that. So I was much upset when I had a phone call from Bob that I'd left out the text disc (a DOS one, that - three different formats in the pack!).

Fortunately, I now keep copies of everything, and I didn't have too much trouble in duplicating it and taking it down to Gloucester, except the cost, inconvenience, and worst of all, the time. This, of course, produced a knock-on. Bob always edits to fit the space available. This time, due to my stupidity, he was short of time and left out a piece he ought to have put in and put in a piece he ought to have left out. So I must try and put it right now. So; you remember Frank Evans, who, last month, sent me a little algorithm for LCM? Somehow, at the end, he got cut off in his prime. And we printed that it was all a lot of Limericks! What I really meant to say was "Thanks, Frank". The Limerick bit referred to all those poor individuals who have been wading through all these pages to find something creative to do with their SAMs, but still don't know what to do. J.Smith of Leeds apologises that he couldn't make this poetry competition in time for the closing date, but sends a SAMjob specially for us at 'Short Spot'. It contains four verses, each of similar type, in DATA statements. The computer chooses lines randomly; (well, not quite: chooses one of the first lines at random, then one of the second, and so on). The end result is a five-line poem. They all bear a certain similarity, but they're all different. The comment that "as a collection of Limericks: well, they're awful, but as a SAM advertising stunt; it's great" belongs here, of course. Nice one; many thanks, Mr Smith. Here it is:-

```
 10 REM SAM POETRY by J.SMITH
    Oct.93
 20 DIM A$(5,4,32)
 30 FOR F=1 TO 5: READ B$: LET
    A$(f,1)=B$: NEXT F: FOR F
    =1 TO 5: READ C$: LET a$(f
    ,2)=C$: NEXT F: FOR F=1 TO
     5: READ D$: LET A$(f,3)=D
    $: NEXT F: FOR F=1 TO 5: R
    EAD E$: LET A$(f,4)=E$: NE
    XT F
 40 CLS : PRINT TAB 5; "SHEER
    POETRY by SAM": PRINT TAB
    5;"____________________": P
    RINT ''
 50 FOR f=1 TO 5
 60 LET a=INT RND(4)
 70 IF a=0 THEN GO TO 60
 80 IF a=1 THEN PRINT 'a$(f,1)
 90 IF a=2 THEN PRINT 'a$(f,2)
100 IF a=3 THEN PRINT 'a$(f,3)
110 IF a=4 THEN PRINT 'a$(f,4)
120 NEXT f
130 PRINT ''''"Another verse?
    Press Y:Q to Quit"
140 PAUSE
150 IF INKEY$="Q" OR INKEY$="q
```

```
" THEN STOP
160 GO TO 40
170 DATA "A friendly computer
    called SAM","Read now how
    famous I am","It's one of
    my views","I'm easy to use
    ","Buy me as soon as you c
    an"
180 DATA "A versatile fellow i
    s SAM","You know what a gr
    eat guy I am","To give me
    my dues","I'm not hard to
    use","Try me, become a SAM
     fan"
190 DATA "I'm a computer calle
    d SAM","A wonderful thing
    that I am","You'll not sta
    nd to lose","I'll not blow
     my fuse","To please you I
     know that I can"
200 DATA "If you wish for a br
    and new SAM","To own one y
    ou certainly can","Don't h
    ave the blues","Three chee
    rs it's good news","Own me
     right now if you can"
```

I also have a reply from Frank Evans, who writes somewhat astringently that his intent was brevity rather than obfuscation. Frank has produced a cornucopia of goodies for me, including a large spirograph program to end all spirographs. Although I've finished this topic now, I feel that his comments on LCM and HCF are well worth printing, so they're below. If you want a copy of Frank's Spirograph program, do leave me a message or drop me a line. Frank, has also sent me a five-liner commenting on my unfavourable comparisons of his code with a Limerick.

"There was a young man from Pershore, Who didn't know his AND from his CURSOR. Was this John Wase? That veritable case. Whose life is one round of pleasure!!!"

Well, as doggerel, I guess it wins the big prize. More importantly, Frank mentions that the method used to find the LCM is first to find the Highest Common Factor (also known as the Greatest Common Divisor) by direct division. This is a recursive process, dividing the lowest number into the larger until zero is reached.

```
Divisor ) Dividend ( Quotient
          Product
          Remainder
```

For the second and subsequent passes, the Remainder becomes the new Divisor, and the Dividend is the old Divisor. When there is no remainder, the process stops, and the last Divisor is the Highest Common Factor.

The Lowest Common Multiple is found by:-

HCF*(WHEEL/HCF)*(RING/HCF)

Many thanks, Frank.

Now back to the present. This month several people have been delving into the Spectrum's past history. One reader has sent me several old volumes which I'll refer to later; sadly I've only had time to skim through parts of them. Another, Mr L.G.Baumann, has been delving into odd corners of the ROM. If you look at the code in detail, you will find that there is a 'trap' there which does something to the display file: it's only a few lines of code and is not normally in use. However, you can call it, and Mr Baumann has found that provision is made to extend the normal 32 column screen to a 36 column screen. I'm not an expert: I don't know how this works. The arrangement of the display file is odd, sandwiched right at the start of things. You can have arrangements alike 42 letters across the screen by using character squares only 7 pixels across, but apparently this is a genuine screen, so I guess that since the display file space is fixed (i.e. the number of pixels) there's fewer than the normal 22 rows; can anyone clever sort it out for me, please? Mr Baumann suggests that this could have been for a further Spectrum that never came out, but I don't honestly think so; my view is that it was written into the ZX81 ROM as a possibility, and no-one took it out when the code was ported to the Spectrum.

Unfortunately, a direct address to be called was not included, so the short program from Mr Baumann builds up the Routine into Memory at Address 60000 from the appropriate addresses in the ROM, and then prints an example as well as the codes used. Mr Baumann can't think of many uses for this sort of screen, though he does mention that 36 columns should be much more useful for tabulations, as it is divisible by 2, 3, 4, 9, 12 and 18. My mind's like our kitchen - the old one was ripped out today by workmen, sink and all, and I'm not thinking right. But all you intelligent readers should be able to incorporate this in your own routines and come up with some ideas for 'Short Spot'. So let's have a little compo; come up with the best idea for using this routine in less than 20 lines of Basic.

Here it is:-

```
 10 CLEAR 59999: POKE 60000, PE
    EK 10645; LET r$=""
 20 FOR f=60001 TO 60010: READ
    a
 30 POKE f,PEEK a: LET r$=r$+CH
    R$ PEEK a
 40 NEXT f
 50 DATA 3155,468,2082,237,5458
    ,11387,1041,514,2289,12183:
     REM the ROM codes 60 PRINT
     "This line prints a normal
     length"
 70 RANDOMIZE USR 60000
 80 PRINT "And this newer line
    is 36 characters": PRINT 'r
    $
8999 STOP
```

Many thanks, Mr Baumann.

Next to Lee Willis. You remember, Ettrick Thomson slapped my wrists over him, because I inadvertently included a snippet which duplicated his offering previously. Lee, too, mentions this, though he clearly doesn't hold it against me, for he's sent some more stuff. One is a program for solving quadratics. Simply enter x squared, x and the constant, and the program does the rest.

```
 10 MODE 4: CLS #: CSIZE 8,8
 20 POKE UDG CHR$ 144,96,16,32
    ,112,0,0,0,0: POKE UDG CHR
    $ 145,16,16,124,16,16,0,12
    4,0: POKE UDG "-",0,126,0,
    0,0,0,0,0: BLOCKS 0
 30 PRINT '" INPUT NUMBER OF X
    ";CHR$ 144;"s"
 40 INPUT #2; AT 3,1; LINE a$
 50 LET A=VAL a$
 60 LET Equ$=STR$ INT a+"X"+CH
    R$ 144
 70 CLS 1
 80 PRINT '" INPUT NUMBER OF X
    's"
 90 INPUT #2; AT 3,1; LINE a$
100 LET B=VAL a$
110 LET Min=B<0
120 LET Equ$=Equ$+("-" AND Min
    )+("+" AND NOT Min)+STR$ (
    ABS B)+"X"
130 CLS
140 PRINT '" INPUT THIRD VALUE
    "
150 INPUT #2; AT 3,1; LINE A$
160 LET C=VAL A$
170 LET MIN=C<0
180 LET Equ$=Equ$+("-" AND Min
    )+("+" AND NOT Min)+STR$ (
    ABS C)
190 CLS
200 PRINT AT 0,0;"EQUATION TO
    SOLVE :-"
210 PRINT AT 2,(24-LEN EQU$)/2
    ;EQU$
220 LET D=B-B-B
230 LET E=B*B
240 LET F=A*C
250 LET G=2*A
260 LET H=E-(4*F)
270 IF h<=0 THEN PRINT "UNABLE
     TO SOLVE QUADRATIC !": ST
    OP
280 LET H=SQR H
290 LET EquTwo$=STR$ D+CHR$ 14
    5+STR$ H
300 PRINT AT 5,(26-LEN EquTwo$
    )/2;EquTwo$
```

```
310 PRINT AT 6,(26-LEN EquTwo$
    )/2;STRING$(LEN EquTwo$,"_
    ")
320 PRINT AT 8,(26-LEN STR$ G)
    /2;G
330 LET SOne=(INT ((((D+H)/G)*
    100)+.5))/100
340 LET STwo=(INT ((((D-H)/G)*
    100)+.5))/100
350 PRINT AT 9,0;"X=";SOne; AT
     11,0;"or"; AT 13,0;"X=";S
    Two
```

There's a tale about this, too. I'd not long had the disc when there was a telephoned plea from Lee "Send me a copy of my disc, quick, perleaze". It seems that it was part of a project or something, and Lee had sent me one and kept the duplicate. And promptly corrupted it. By the time he reads this, he should have eight copies. We all know SAM's bad habits, though, don't we. For crying out loud, ALWAYS keep things in duplicate on two separate discs. (And don't mention 'Word for Windows' to me: it sets me off in an enormous fit of temper).

The next thing Lee has sent is a snappy HCF job: no, not a complete Spirograph: just the HCF bit. So we'll add that...

```
10 IF x>y THEN LET z=x,x=y,y=
   z,z=0
20 LET LCM=((x*y) AND (y MOD
   x))+(y AND NOT (y MOD x))
```

And finally, I've got a real one-liner. A proper Short Spot for April. Here's 'Fancy-End'.

```
POKE ((DPEEK 23200)=16384),255,2
   55
```

What's it do? Do? Why, put it as the last line of your program and RUN the program. The listing will mysteriously vanish from your computer, leaving you with a neat OK message at the end. Nice one, Lee.

Lee also includes a load of hidden files with a title screen with a scrolly demo I can't really read and lots and lots of music. Fine. It would have been better without, though; instead, a letter would have been great, so that I don't have to squint at a SECOND screen all the time, while I'm typing on the first. And by the same token, make sure your address is on the paper.... Lee nearly didn't get his project discs....

Many thanks, Lee...

Now, you will remember last month I cut Ettrick Thomson off short. So here's some more. Ettrick mentions that the equation that Ted Cooke-Yarborough dealt with in February's 'Short Spot' lends itself to all sorts of interesting computer programs. One of the most interesting is a Bifurcation diagram or Feigenbaum diagram (called after Mitchell Feigenbaum) for the equation. The program plots a graph with p as x co-ordinate and x as y co-ordinate. For each of the 256 values of p, 100 successive values of x are plotted: not the first 100, but the second hundred; the idea being that the values plotted are independent of the initial value (chosen arbitrarily as 0.5), and show what happens in the long run. The graph covers the values of p from 2.75 to 4, with graduations every 0.25. This range of p has been chosen to show the interesting phenomenon that Ted mentioned in February. For p<1, x tends to zero, and for 1<p<3 to 1-1/p, and so in the range 2.75 to 3, the graph goes from 0.636.. to 0.666.., but in the critical region, just less than 3, which Ted mentions, it evidently has not settled down after 100 iterations. Immediately above 3, x alternates between 2 values; the diagram does not show the alternation; just the fact that there are 2 values. Then, as p increases, there are 4 values, then 8 values, then 16, then.....

But, round about p=3.57, chaos sets in, and continues until p=4, with occasional returns to order, notably round about p=3.83, where x has 3 values, then 6, then... then more chaos.

The program can be altered to cover any range of p,x; one easy alteration is to change the first statement of Line 110 to `LET p=3.5+u/510`; the graduations drawn by Lines 10-30; have to be interpreted as 3.5, 3.6, 3.7,... 4.0. This shows the chaotic region in greater detail. The program will run either on Spectrum or SAM.

```
  5 REM Feigenbaum diagram
  7 REM Ettrick Thomson
 10 FOR p=2.75 TO 4 STEP 0.25
 20 PLOT (p-2.75)*204,0: DRAW
    0,172
 30 NEXT p
 40 FOR x=0 TO 1 STEP 1/4
 50 PLOT 0,x*172: DRAW 255,0
 60 NEXT x
100 FOR u=0 TO 255
110 LET p=2.75+u/204: LET x=0.
    5
120 FOR i=1 TO 100
130 LET x=p*x*(1-x)
140 NEXT i
150 FOR i=1 TO 100
160 LET x=p*x*(1-x)
170 PLOT u,172*x
180 NEXT i
190 NEXT u
```

There's still more left for next month, too. Many thanks, Ettrick,

While we're on the subject of chaos (like our dining room, piled high with boxes full of what ought to be in kitchen cupboards if there were any), I've also got a little bit of chaos from W.T.Buxton of Sale in Cheshire. Bill writes with a somewhat similar Basic program which can be reduced to six lines if needed, and which demonstrates Hunt and Johnson's formula numerically.

With the input of any given value of X (0 to 1) and P (1 to 4), this program calculates the numerical value of X obtained in the one n+1 calculation which is then used for X in the next n calculation, The resultant values are then listed to infinity and each one can be compared with its neighbours without having to do any more calculations. Bill finds that examining these values gives one an interesting insight into the complex and chaotic nature of the oscillations between calculations, suggesting ***FORMAT*** readers will want to investigate.

Well, I guess that Ettrick has done a fair amount of investigation already, but we're always game for more, aren't we...

```
 10 REM CHAOS
 20 REM Numerical demonstratio
    n by W. T. Buxton
 30 INPUT "INPUT X(0 TO 1)=";X

 40 INPUT "INPUT P(1 TO 4)=";P

 50 PRINT "IF X.n+1=P*X.n*(1-X
    .N)": REM Equation of HUNT
     AND JOHNSON SEE IEEE Spec
    trum Nov 1993
 60 PRINT
 70 PRINT "FOR P=";P,"X=";X
 80 PRINT
 90 PAUSE 50
100 PRINT " Then X.n+1 = ";P*X
    *(1-X)
110 PRINT "==================
    ======= ======"
120 LET X=P*X*(1-X)
130 GOTO 70
```

Many thanks, Bill.

Let's end with the usual plea to send all your bits and pieces to:-

John Wase,
Green Leys Cottage,
Bishampton,
Pershore,
Worcs,
WR10 2LX.

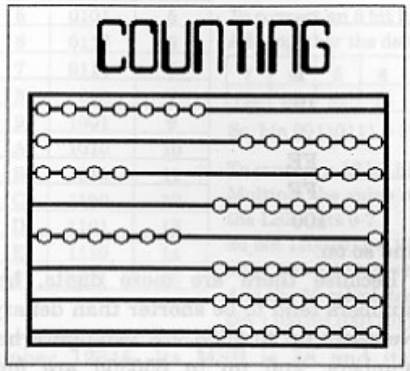
Without them, this column wouldn't get done and there would be a very large hole in ***FORMAT***.

Thank you.

MACHINE CODE WITHOUT THE TEARS

Part 25.

By:- Carol Brooksbank.



We normally calculate and count in the decimal - or denary - system, using 10 digits 0-9. This is the system which comes naturally to humans, because we are born with five digits on each hand, and we all learn to count on our fingers.

But there are other systems of counting, and they are frequently used in the computer world. Binary uses only 2 digits, 0-1. Hex uses 16 digits, 0-9 and the letters A-F. Octal uses 8 digits, 0-7.

To understand how these other systems work, it is vital to understand what we do when we count in denary. Normally we don't stop to consider what we are actually doing when we count, say from 1 to 100. We know what to do so we do it. But let us analyse the process.

We have 10 digits, so counting from 0-9 is no problem:-

0
1
...
8
9

But now we have run out of digits, so we start a new column to the left of the one we have been using, put 1 in it, start back at 0 in our original column and carry on:-

10
11
...
18
19

When we run out of digits again, we add 1 to the left column, and start again at 0 in the right:-

20
21
...
29
30

This goes on until we run out of digits again in both columns, when we repeat the process - add another column to the left, put 1 in that, change the ones on the right to 0 and carry on as before:-

99
100
101

and so on. Every time all the columns are at 9, we can invent a new one at the left. There is no limit to the size of number we can express using the system, because we can use as many columns as we like.

The process is exactly the same in all the other systems, the only difference being the number of digits available. In binary, where we have only 2 digits, we have to bring in new columns very

frequently, and so the numbers are generally longer than denary numbers:-

BIN	DEN
0	0
1	1
10	2
11	3
100	4
101	5
110	6
111	7
1000	8

By the time we have reached denary 8, binary is a 4-figure number. The usual convention is to express all binary numbers up to 255d as eight digit numbers, so 7d is 00000111bin.

So why use binary? Binary is the only system your computer understands. All its memory locations are capable of holding 1 number between 0-255d, 00000000-11111111bin. You can think of each memory location as a bank of 8 tiny switches, switched on or off according to the bit status of the binary form of the number held there. The current can only flow through the 'on' switches (set bits), and it is the combination of switch settings which causes the processor to perform various operations. That is what you are doing with machine code programming - setting the banks of switches in the right pattern and order to get the result you want.

If you program in Basic, the Basic interpreter converts the Basic commands to a series of machine code binary instructions, and the operating system can also convert denary input, because that is what most humans are likely to use. Binary is much too cumbersome for everyday use.

Hex - short for hexadecimal - is another counting system, working in exactly the same way as the others, except that there are now 16 digits, the letters A-F being used as numerical digits in addition to 0-9. The use-the-digits-up-then-bring-in-a-new-column system is used again.

0
1
...
E
F
10
...
FE
FF
100

and so on.

Because there are more digits, hex numbers tend to be shorter than denary. Numbers up to 255d are two-figure hex numbers, and up to 65535d are four figure hex numbers. We usually express hex numbers as two or four figures - 13d = 0Dh, 16384d = 4000h. To avoid confusion between numbers which look the same in both conventions, we reserve the usual way of saying numbers for the denary system. 10d is 'ten' 10h is 'one-oh-hex'.

So why use hex? Its number base is 16, 2*2*2*2, so there is a closer relationship between the binary numbers your computer understands and hex notation, than there is between denary and binary, but it is less cumbersome than binary. The 16 hex digits cover every possible combination of bit settings in a 4-digit binary number. (see Table A).

It is much easier to see the relationship between a number's MSB and LSB in hex than in denary. You know that the computer stores a number above 255 in two locations, or in a double register, in the form of MSB=INT(n/256), and LSB=n-(256*MSB). Take the denary

TABLE A. HEX-BINARY-DENARY EQUIVALENTS.

HEX	BIN	DEN
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

To convert between HEX and BINARY.
Convert each hex digit to the 4-digit binary number, or each group of 4 binary digits to the equivalent hex digit.
e.g. CBh = 11001011bin
bin 1100101100110111 = 1100 1011 0011 0111
C B 3 7

To convert an 8 bit binary number to denary.
Add together the denary values for each set bit.

7	6	5	4	3	2	1	0	Bit Number
128	64	32	16	8	4	2	1	Denary Value

So, bin 00110111 = 32+16+4+2+1 = 55 denary.

To convert a 16 bit binary number to denary.
Multiply the value of the MSB (bits 8-15) by 256 and add the value of the LSB (bits 0-7.
So bin 11001011 = 128+64+8+2+1 = 203
203 * 256 = 51968 when this is the MSB.
Add the 55 calculated above for the LSB, and
bin 1100101100110111 = 51968+55 = 52023.

number 12345. Its MSB is 48 and its LSB is 57. There is no obvious connection between MSB and LSB unless you do the calculation. But in hex it is different. The hex equivalent of 12345d is 3039h. The MSB is 30h and the LSB 39h. You can see at a glance what a number's MSB and LSB are. Check these hex numbers in Tables A and B to see for yourself that they are the same as the denary equivalents.

If you like the look of hex - if it intrigues you and you would like to learn it you will find that in the end it makes machine code programming easier. The books by William Tang or Toni Baker, which I mentioned last month, will help you learn it.

If you don't like the look of it - if it seems fiddly and unneccessary, it will only put you off machine code altogether if you force yourself to learn it. If you have two programmers doing the same job, one fluent in hex and one fluent in denary, each working in the convention they like, the one working in hex will usually get on faster. But if you have two programmers, one working in hex because he thinks he ought to but not really happy with it, and one working in denary, the one working in denary will win hands down.

So don't let anyone tell you you must work in hex. Unless it makes life easier for you, stick to denary. When you come across hex in listings, your assembler will almost certainly do the conversion for you. If not, Table A and a calculator will let you convert hex-denary.

Table A shows you the relationship between hex, denary and binary. We have used binary quite a lot in this series already. Generally, if you need to think about individual bits rather than the number as a whole, it is easier to work in binary. But you can enter the bit patterns you need as binary numbers and you don't then need to worry about the denary equivalent of the number. If you need to convert between conventions, Table A will help.

The other convention used in

computing is octal. The columns system is exactly the same as in all the other counting systems, but now there are only 8 digits, 0-7. Octal digits cover every possible combination of bits in a 3-figure binary number.

Octal is not, as far as I know, used in the Sam/Spectrum world very much - I have never come across a Sam/Spectrum machine code listing in octal. But it is widely used in the computer world as a whole, so you should be aware of it. Some Sam/Spectrum assemblers will let you enter octal numbers if you wish.

All assemblers will let you enter denary numbers as they stand, but will usually require a prefix for any other convention. Mine (ASTRUM+ for the Spectrum - I write all my machine code programs on the Spectrum, even the Sam ones) uses the following forms:-

nnnn Denary (decimal) number
#nnnn Hex number
_nnnn Octal number
@nnnn Binary number
"abc" ASCII codes for characters within " " are stored.

Your assembler may be different. & and H are often found as hex prefixes. B or BIN may be used for binary. Some assemblers let you switch into hex mode and then enter hex numbers without prefix and need a D before denary numbers. Your assembler handbook will tell you what you need to do.

With the help of the tables I have given you here, and your assembler, you should be able to use any machine code listings you find in print.

So there we are, the end of the series. But for you, it is only the end of the beginning. Because I hope you will be striking out on your own, writing your own routines and programs, and sharing them with the rest of us by writing them up for ***FORMAT***, of course.

This series has run for well over two years, since August 1991, in fact - we have missed one or two months when there have been special Christmas or Anniversary issues of ***FORMAT***. Several people have been kind enough to congratulate me on "Machine Code Without The Tears", but the real congratulations belong to you, who have stuck with me all this time and now know the fundamentals of machine code. Don't try to remember it all. Look back to previous articles to refresh your memory. The real secret of being a good machine code programmer is not knowing all there is to know, but knowing where to look it up when you need it.

What am I going to do now? For a start off I am going to ride off into the sunset for a bit, before our beloved Editor finds some other two-year project to keep me out of mischief. It is spring 1993 while I am writing this. So, while you slave over a hot keyboard, I plan to sit in a deckchair this summer, sip something long and cool, and hope it will be some time before the phone rings and a voice says "Bob here. I've had an idea..."

'Bye for now.

BASICally Speaking...

Part 6.

By:- David Finch.

Welcome. Firstly I would like to make a correction. Line 60 of the program which we typed in last month is slightly wrong. The purpose of this line is to check that we have a filename, which is entered in another part of the programme. The scrabble rack will have ten blanks if no name has been entered, so we don't want to save. Add line 5:-

```
5 DIM b$(10)
```

This creates a blank string called b$, with ten blank spaces in. We can compare this to fname$ (or f$ for Speccy) to see if we need to save.

Change line 60 to:-

```
60 IF fname$<>b$ THEN SAVE fna
   me$ SCREEN$ (SAM)
60 IF f$<>b$ THEN SAVE f$ SCRE
   EN$ (Speccy tape)
60 IF f$<>b$ THEN SAVE d1;f$ S
   CREEN$ (PLUS D,DISCiPLE)
```

Secondly, it has been brought to my attention that GOSUBs within GOSUBs work perfectly well (thanks to Ettrick Thomson and Roy Burford). In my defence, your honour, I used GOSUBs within GOSUBs when writing a program for a friend once. The Spectrum crashed every time the program ran and so I've blamed GOSUBs ever since. So folks, GOSUB as much as you like - within reason of course. The pie chart program will go ahead as previously written. Sorry for any inconvenience, normal service will resume shortly.

Now, back to business. I would like to recap on what you should have so far with the input procedure (note, this is not inputdata, that will come later).

SAM

```
560 DIM n$(z)
570 LET c=1
580 DO
590 PRINT INVERSE 1;AT y,x;n$
600 PRINT INVERSE 1;PEN 2;AT y
    ,x+c-1;n$(c)
610 GET k$
620 LET n$(c)=k$
630 LET c=c+(c<>z)
670 LOOP
```

Spectrum users make necessary changes to lines 580,600 (PEN to INK),610,670 as instructed last month.

Instead of typing in all the LETs before we run the above, all the SAMsters can turn it into a procedure:-

```
540 DEF PROC inpu x,y,z
690 END PROC
```

Now you can type: inpu 1,1,20 to try the procedure as you did last month. Spectrum programmers, I'm afraid you have to type in your LETs. You can add:-

```
540 REM inpu
```

for neatness. It is not essential but makes the program easier to follow. You may also like to add `539 STOP` to prevent the 'procedure' from running accidently after running line 530. This shouldn't happen, but taking precautions like this prevent unexpected factors, or modifications to the program, from causing further problems.

```
690 GOTO r
```

This is like END PROC. The variable r must be set before we GOTO this part. We could use GOSUB and RETURN.

There is another problem with the

procedure. You cannot delete or move the cursor back and forward. We need to sense if the key pressed (k$) is the delete key, or the arrow keys. We use the CHR$ command for this: Every letter is represented in a code called ASCII by a number. For example `PRINT CHR$ 65` does the same as `PRINT "A"` since 65 is the ASCII code for capital A. You can make a condition to sense if the A key has been pressed using `IF k$=CHR$ 65 THEN...` Similarly, you can sense if the delete keys or arrow keys have been pressed since their codes (on the Spectrum and SAM), are 8, 9 and 12 for LEFT, RIGHT and DELETE respectively.

Change line 620 to 650, and line 630 to 660, then enter the following.

```
620 IF k$=CHR$ 8 THEN LET c=c-
    (c<>1): LOOP
630 IF k$=CHR$ 9 THEN LET c=c+
    (c<>z): LOOP
640 IF k$=CHR$ 12 THEN LET n$(
    c)=" ": LET c=c-(c<>1): LO
    OP
```

(Spectrum users type GOTO 590 instead of LOOP.)

The first line decreases the cursor by one, so long as it isn't the first space, when the left arrow key is pressed (if this confuses you, see the last bit of part 5). The second line detects the right key and increases the cursor if it is not the last (there are z spaces). The third makes the current cursor position blank (deletes it) and takes the cursor back one as in line 620.

Our procedure must now sense when RETURN (ENTER) is pressed and finish the procedure. At the moment the procedure will loop infinitely. We need to test if RETURN is pressed, and exit the loop if it is. The ASCII code for the RETURN is 13, so we must change line 670 to:-

```
670 LOOP UNTIL k$=CHR$ 13
```

Or for Spectrum users (you are a nuisance!):-

```
670 IF NOT (k$=CHR$ 13) THEN
    GOTO 590
```

I was kidding about you being a nuisance, sorry folks. They say sarcasm is the lowest form of humour and I'm wasting precious ***FORMAT*** ink while I should be telling you that the next procedure we are going to do is 'setup'. This procedure allows the user to input the title of the pie chart, the filename it is to be saved as, and the number of sectors (pie wedges) which are needed. The procedure is quite simple. It clears the screen with our clr procedure, prints the titles for the information required then allows the user to type in the information.

```
330 DEF PROC setup (SAM)
330 REM setup (Spec)
340 clr (SAM)
340 LET z=350: GOTO 700:  REM c
    lr (Spec)
```

Note that this sets variable z to 350 so that clr can GOTO 350 when it has finished. Procedures do this automatically for SAM users.

```
350 PRINT INK 2; AT 1,13;"SET
    UP"
360 INK 1
```

SAM will convert INKs to PENs for you automatically, since SAM uses the PEN command instead.

```
370 PRINT AT 5,2;"How many sec
    tors?"
380 PRINT AT 8,2;"Title?"
390 PRINT AT 12,2;"Filename to
     save?"
400 PRINT AT 5,1; INK 4;"*"
410 DO: REM Spectrum users omi
    t
420 INPUT #2; AT 5,19;sec (SAM
    )
420 INPUT sec (Spec)
```

Of course, that the SAM needn't input from the lower screen area, by adding #2;

to the start you instruct it to input from the main screen (like a PRINT command), AT the specified print position. This is not available to Spectrum owners, tut tut Sir Clive!

```
430 LOOP WHILE sec<>INT sec OR
    sec<1 (SAM)
430 IF sec<>INT sec OR sec<1 T
    HEN GOTO 420
```

The above line ensures that sec (the number of sectors) is an integer (whole number). The function INT converts a number to the same number without any decimal fraction. If the INT number is not the same as the inputted number then the inputted number can't be a whole number. It is important that sec is whole because we can't have four and a half sectors, for example. It also causes the user to input the number again if it is less than one (ie zero or negative).

```
440 PRINT AT 5,1;" " 450 PRINT
    AT (,1;INK 4;"*"
```

These stars print beside the titles to indicate which is being inputted. Line 440 erases the previous one, and 450 prints the next.

```
460 inpu 1,9,30 (SAM)
460 LET x=1: LET y=9: LET z=30
    : LET r=470: GOTO 540: REM
     inpu (Spec)
```

Do you see why procedures are easier? This allows you to type in the title, maximum length 30 characters. It is produced in the form of n$.

```
470 LET t$=n$
```

t$ is the string used for the title. We can't keep it as n$ because it will be overwritten by the next use of the inpu procedure, which is next, to enter the filename.

```
480 PRINT AT 8,1;" "
490 PRINT AT 12,1;INK 4;"*"
500 inpu 2,13,19 (SAM)
500 LET x=2: LET y=13: LET z=1
    9: LET r=510: GOTO 540: RE
    M inpu (Spec)
510 LET fname$=n$ (SAM)
510 LET f$=n$ (Spec)
520 PRINT AT 12,1;" "
530 END PROC (SAM)
530 RETURN (Spec)
```

After we have set up the pie chart we need to input the data, so the next procedure we are going to input is called inputdata. Your computer already knows how many sectors are required (the variable sec). We are going to store the numbers in an array. An array is a way of storing numbers like in a table. You can imagine it as being a sheet of paper divided into rows and columns, with one number in each box. You treat each box like a separate variable. You can also have pages of these tables, and then books, then volumes (the libraries I suppose). If you don't understand yet, read on because it may get clearer.

Before we had a variable to store numbers, for example, z. The variable z could hold one number. So far, if we have several numbers, we had to use several variables. However, if you DEFine the array, you can store more numbers in the same variable, for example:-

(Don't include this in your program)

```
10 DIM z(10)
```

This creates the ARRAY z, which can hold ten numbers. If you want to use the first of the numbers, you use z(1) like you would use any other variable. If you want to access the second number you use z(2), and so on.

If you imagine the above system as a list of numbers, you can also create a table of numbers, for example:-

```
10 DIM z(5,10)
```

Creates an array, z, with 5 rows and 10 columns, holding 50 numbers. If you wanted to use the number in row 2, column 4 you can use z(2,4) like any other variable.

You can add another dimension if you wish, ie `DIM z(5,10,8)`. Note that when you DIM an array you also set all the values within to zero, so you must take care never to DIM an array which already exists and contains important numbers.

For our program I want to create an array which contains six numbers per sector. This array will contain all the information required to create the pie chart.

```
80 DEF PROC inputdata (SAM)
80 REM inputdata (Spec)
90 DIM n(sec,6)
```

So we have an array called n, and 6 entries for every sector. The first of those six numbers will hold the number that the user enters. We need to allow the user to enter these values, and change them if necessary. The user must also specify which of the sectors he/she wants 'off set' (see the BASICally Speaking sector in the example chart last month). Any number of sectors can be off set.

If we let the sector we are currently editing be m, we want to edit the value of sector 1 first: -

```
100 LET m=1
```

We will now loop round while we enter all the numbers.

```
110 DO (SAM only)
120 clr (SAM)
120 LET z=130: GOTO 700: REM c
    lr (Spec)
130 PLOT 0,130: DRAW 255,0 (SA
    M)
130 PLOT 0,138: DRAW 255,0 (Sp
    ec)
140 PRINT AT 1,1;"Sector numbe
    r:";m;" of ";sec
150 PRINT AT 2,1;"Current valu
    e:";n(m,1)
160 PRINT AT 3,1;"Off set :"+(
    "On" and n(m,2))+("Off" an
    d NOT n(m,2))
170 PRINT AT 5,1; INK 2;"Type
    in the new value"
180 PRINT AT 6,1;"and press RE
    TURN."
190 PRINT AT 8,1;"Then/Or:"
200 PRINT AT 10,1;"Press + or
    - to edit the next"
210 PRINT AT 11,1;"or previous
     sector."
220 PRINT AT 15,1;"Press : whe
    n finished all."
230 PRINT AT 13,1;"Press ; to
    togyle offset."
240 inpu 1,17,30 (SAM)
240 LET x=1: LET y=17: LET z=3
    0; LET r=250: GOTO 540: RE
    M inpu (Spec)
250 IF k$="+" THEN LET m=m+1:
    IF m>sec: LET m=1: END IF:
     LOOP
260 IF k$="-" THEN LET m=m-1:
    IF m<1: LET m=sec: END IF:
     LOOP
270 IF k$<>":" AND k$<>";" AND
     n$(1)<>" " THEN LET n(m,1
    )=VAL n$
280 IF k$=";" THEN LET n(m,2)=
    NOT n(m,2)
290 LOOP UNTIL k$=";" (SAM)
290 IF k$<>":" THEN GOTO 120 (
    Spectrum)
300 clr (SAM)
300 LET z=310: GOTO 700: REM c
    lr (Spec)
310 PRINT AT 1,1;t$
320 END PROC (SAM)
320 RETURN (Spec)
```

The procedure prints the instructions to the screen and then uses inpu to take in information. The next five lines check k$, which holds the last key pressed (n$ holds the string entered). So if the user presses + then the program increases m by one (m is the sector number which e is editing). Note that it checks if m is a number greater than the number of sectors (sec), and if so, it LETs m=1. Similarly for the - key, and if you attempt to press - when on the first sector it will take you up to the last (m=sec).

If either + or - have been pressed it will have looped, so now we needn't worry about those keys. If neither the : or the ; keys have been pressed we must be entering the value of the sector. We have

a problem here folks. We have the number which we have entered in n$, and the value needs to go in n(m,1), a numerical array. You must use the function VAL, which converts a string containing a number to a form which can be used like a number. The opposite function is the command STR$, which converts a variable or numerical array into a form which can be used like any other string.

When ; is pressed, line 280 toggles the value in n(m,2) (remember m is the sector currently being edited) from one to zero, or vice versa. You should remember that NOT changes a true to false and a false to true, and that true is represented by one, false by zero.

The user should press : when all the values have been entered, and line 290 goes back for another input if it hasn't been pressed.

We have another problem. Suppose the user enters text instead of a number. Our procedure will try to VAL some letters, and the poor computer can't cope with that. It will give an error message. We need to prevent the user from entering letters and the easiest way to do that is within the inpu procedure. The inpu procedure needs to know when you are entering numbers (as above), or when you need to enter text (when it is used for the title for example).

```
550 LET u=(x=1 AND y=17)
```

This sets the variable u to true when the input bar is at the point specified. We will restrict numbers only when u is true. I have done this to demonstrate some BAD programming. Suppose you wanted to modify this program. You may need to use the inpu procedure. If you happened to place your input bar at the same place you would find that you could only enter numbers, if you want to or not.

A better solution would be to LET u=1 before you run the inpu procedure for numbers, then let it equal zero again when it has finished. Can you see why?

When we want to enter text we want to allow all letters, numbers and symbols to be entered. These all lie between the ASCII codes 32 and 121 inclusive. We can text if k$ is between these codes like we would test variables and numbers.

Delete lines 650 and 660 then enter:-

```
650 IF (NOT u AND (k$>CHR$ 31
    AND k$<CHR$ 122)) THEN LET
      n$(c)=k$: LET c=c+(c<>z)
```

The part which reads (k$>CHR$ 31 AND k$<CHR$ 122) is only true if the ASCII code for k$ is between the 31 and 122 exclusive.

```
660 IF u AND ((k$>CHR$ 47 AND
    k$<CHR$ 57) OR k$=".") THE
    N LET n$(c)=k$: LET c=c+(c
    <>z)
```

This allows only numbers or the decimal point when u is true.

Also, if our inputdata procedure is to detect he +,-,; and : keys ten this procedure must finish when any of those keys are pressed. Change line 670 to:-

```
670 LOOP UNTIL k$=CHR$ 13 OR (
    u AND (k$="-" OR k$="+" OR
    k$=":" OR k$=";"))
```

Also, for neatness, we will PRINT the result without inverse before the procedure finishes.

```
680 PRINT AT y,x;n$
```

At line 690 you should have the END PROC or GOTO.

Next month we are going to manipulate the data inputted, and use it to draw the chart.

See ya'.

Uni-Dos Corner

By:- Henk van Leeuwen. Edited by:- Adrian Russell.

UNI-DOS is the advanced ROM/DOS system for both the DISCiPLE and PLUS D marketed by S.D.Software. This series of articles helps you to write CREATE files - a means of adding new commands and functions to UNI-DOS which can be used in your own Basic programs.

The CREATE files, when loaded, are stored in the Basic memory space and they must be written to be relocateable.

When you have a big Basic program and you may want to SAVE just a few lines of it to disc. The new commands we build this month will allow you to do just that. Here is the syntax:-

```
EXP <OVER> <start> TO <end> ;D<d
    rive>"Name......"
```

or

```
EXP <OVER> <start> TO <end> ;D<d
    rive>,a$
```

I think the commands speak for themselves but there are some comments in the listing below. I use EXP as the keyword to stand for EXPort - because we are exporting lines that can then be used elsewhere.

Here is the assembler listing:-

```
        ORG 60000

START   DEFB 1 ;only one syntax
        DEFB 185 ;SAVE keyword
        DEFW EXP_LEN

L_EXP   RST 24
        DEFB 115 ;OVER.SYN
        RST 16
        DEFW 7298
;ask for startline number
        CP 204 ;Token for 'TO'
        RET NZ
        RST 40
        RST 16
        DEFW 7298
;ask for last line number
        CP ";" ;test for ';'
        RET NZ
        RST 40
        AND 223 ;capitals
        CP "D" ;use D for drive
        RET NZ
;if wrong reject syntax
        RST 40 ;end of syntax
        RST 24 ;get drive number
        DEFB 120 ;DR.NUM
        RST 24 ;next char " or ,
        DEFB 112 ;SEPARATOR
        RET NZ ;if wrong reject
        RST 24 ;get the filename
        DEFB 123 ;GET.NAME
        RST 24 ;end of part one
        DEFB 13 ;SYN.END
        RST 24
        DEFB 111 ;check.drive
        RST 16
        DEFW 7833
;get last line number
        LD H,B ;replace BC to HL
        LD L,C
        RST 16
        DEFW 6510
;go to Spectrum ROM and get
;address of line number
        JR NZ,ERROR
;addr found so continue
        LD (15893),HL
;replace line address
        INC HL
        INC HL
        LD C,(HL)
;now point to first byte of
;the length & copy to C register
        INC HL
        LD B,(HL) ;copy to B
        LD A,B
        OR C
        JR NZ,CONT_1
;if both zero then line number
```

```
;not in Basic
ERROR    RST 32
         DEFB 141
;issue parameter error message.
CONT_1   INC BC
         INC BC
;also use bytes for line number
;and length line
         INC BC
         INC BC
         LD (15889),BC
;last line length
         RST 16
         DEFW 7833
;get start line
         LD H,B
         LD L,C ;copy BC to HL
         RST 16
         DEFW 6510
;go to spectrum ROM and get
;address of line number
         JR NZ,ERROR
;address found so continue
         LD (15891),HL
;start address to save
         LD HL,(15893)
         LD BC,(15891)
         SBC HL,BC
         LD BC,(15889)
         ADD HL,BC
         LD (15889),HL ;DRAM HDOB
         LD (15893),HL ;DRAM HDOF
         LD A,1 ;directory type
         LD (15877),A
         LD A,"D" ;device type
         LD (15876),A
         LD A,0 ;file type
         LD (15888),A
         RST 24 ;Opens write file
         DEFB 61 ;OPEN.FILE
         RET NZ
         RST 24
;save a 9 byte block as header
         DEFB 44 ;SAVE.HEAD
         LD HL,(15891)
;get start address into HL
         LD DE,(15889)
;get length  into DE
         LD (15274),DE
         RST 24
         DEFB 39 ;SAVE.BLOCK
;saves a block of DE bytes
         RST 24
         DEFB 66 ;CLOSE.FILE
         RST 24 ;end of part two
         DEFB 14 ;COM.END
EXP_LEN EQU $-L_EXP
```

Right, here are the DATA lines to enter in a Basic data-poker program if you don't have your own assembler.

```
 40 DATA 1,185,135,0,223,115,21
    5,130
 41 DATA 28,254,204,192,239,215,
    130,28
 42 DATA 254,59,192,239,230,223,
    254,68
 43 DATA 192,239,223,120,223,112
    ,192,223
 44 DATA 123,223,13,223,111,215,
    153,30
 45 DATA 96,105,215,110,25,32,12
    ,34
 46 DATA 21,62,35,35,78,35,70,12
    0
 47 DATA 177,32,2,231,141,3,3,3
 48 DATA 3,237,67,17,62,215,153,
    30
 49 DATA 96,105,215,110,25,32,23
    6,34
 50 DATA 19,62,42,21,62,237,75,1
    9
 51 DATA 62,237,66,237,75,17,62,
    9
 52 DATA 34,17,62,34,21,62,62,1
 53 DATA 50,5,62,62,68,50,4,62
 54 DATA 62,0,50,16,62,223,61,19
    2
 55 DATA 223,44,42,19,62,237,91,
    17
 56 DATA 62,237,83,170,59,223,39
    ,223
 57 DATA 66,223,14
```

Once you have assembled the code (or poked it in) then save it as:-

```
SAVE d*"EXPlncode"USR 60000,139
```

If you want to know more about CREATE files then there is an excellent booklet published by S.D.Software that explains all the basics. Although it is not essential, it would be very useful if you had it to hand as you worked through the listings we publish.

Next time here in UNI-DOS Corner we will present a similar routine that will allow you to save just the variables from a Basic program.

See you soon.

The HELP PAGE

Edited By:- Ray Bray.

The first letter this month comes from Bernadette Downsland who has recently purchased a Spectrum +3 and is having problems using a printer connected via the parallel printer port. It would seem that the +3 in question is fitted with one of the earlier ROMs which, when printing, can cause extra line feeds, erratic alignment of the left hand margin and duplication of some characters. Fortunately these problems can all be rectified by entering two commands before loading programs which produce printed output; namely POKE 23354,42 and POKE 23355,16. To make these commands work correctly in a program line it may be necessary to add a third command; POKE 23390,20. Mike mentions that he has to use these POKEs before he can LLIST or LPRINT but the versions of TASWORD +3 and MASTERFILE which he has do not require any alteration.

Ron Deeks of Hove uses his SAM fitted with SAMBUS only at weekends and is dismayed to find that each time he switches the computer on he has to reset the time and date. He feels that his use over the weekend should be sufficient to keep the battery charged and concludes that, either the battery is not doing it's job or that the clock is being corrupted during power up. He wonders whether fitting the external power supply for the SAMBUS would keep the clock running. I don't have any figures for how much computer use is needed to keep the battery charged up, but as you say you use it for many hours each weekend, I wouldn't think that is the cause of your problem. It is possible that you have a faulty battery or the battery connections are faulty and, if possible, you should check for this before looking for more serious faults. The external power supply for SAMBUS only supplies power to any peripheral equipment attached to the SAMBUS, therefore it would not help in keeping the battery charged while the computer is switched off. Ron also wishes to know if the TWOUP connector has a clock. The answer is to this is no.

Stephen Cox is having difficulty getting a stable display from his 512K SAM using the SCART socket despite using two different SCART leads (one from ***FORMAT***), and trying two different TV's and a monitor. He concludes therefore that there must be a fault with his computer and wonders if there is some way that he can 'tune' it to produce a stable display. Unfortunately not all TV's and monitors use a standard pin layout for the SCART socket and you might have been unlucky in trying three equipments which do not follow the layout used for the SAM lead. The main problem is usually caused by the CSYNC and CVIDIN pins which are inter-connected on the lead. If one or both of these are in a different position on the TV or monitor then the picture will not stabilize. You say that you borrowed one SCART lead. If this was

being used for running a SAM then you could check your SAM on the lender's TV/monitor. There is no tuning which you can carry out on the SAM video output without the proper test equipment.

Another question concerning displays comes from R.Barton of Rotherham. He has a Sanyo colour data display which has 8 wires coming from a socket at the back. He knows that the black wire is earth, the red green and blue are the screen colour connections, and the brown is the composite sync. He wishes to know what the other three wires (white, yellow and grey) are for. I don't have any details of this display but looking at the various connections on other equipments I would think that they are either CSYNCGND, CVID and CVIDGND or, REDGND BLUEGND and GREENGND. I couldn't guess which wires would be allocated to these functions. Incidentally CVID does not include audio as your letter implies. Another display he owns requires horizontal and vertical sync inputs instead of a composite sync input. He says he has connected the 'comp and the vert' from a Spectrum +2 but cannot stop the display from rolling. Unfortunately, pin 5 which you ascribe to VSYNC is in fact listed in the handbook as being +12V not VSYNC which, even if the display is accepting the horizontal component of CSYNC, is why the picture is rolling. The answer to your supplementary question on whether it is possible to convert CSYNC to HSYNC and VSYNC and CVID to RGB must be yes, as there are microchips which do this in TV's. However, unless you are an expert in electronics I doubt if it is a feasible proposition as the chips require a lot of additional components to match the video circuitry. Having said that there is bound to be a reader who has found a simple way of doing this!

Printers continue to cause a lot of problems for our readers. John Moore has a fault which I haven't come across before. He runs a Citizen 120D printer with his SAM and although he can print to continuous feed paper, every time he tries to LPRINT to single sheets the paper out light comes on and the SAM crashes. One thing which puzzles me is that the paper out detector seems to be working properly with fan-fold paper so the mechanical detector must be working. However, I am not familiar with the Citizen 120 printer and perhaps it has a switch to select tractor feed which causes the light to be isolated. Most printers have paper out detector isolating switches which can be used to bypass this problem, failing this the appropriate printer control command could be used to do the same thing. However, if as John says the SAM crashes rather than just hanging-up, this seems to indicate something else is wrong, but I can't think of anything which would cause this to happen with single sheets but not with fan-fold paper. Has any other reader experienced a similar problem with the citizen printer?

Finally, we have a couple of responses to previous questions we couldn't answer. Gary Rowland had a problem with finding a replacement +3 disc drive last October. He has now written saying that he has found a supplier of a suitable drive which was originally intended for the CPC computer (Part No.EME 157). Installation is simply a matter of unplugging the old drive, transferring the mounting bracket to the new drive and plugging it in. The drives, which have a different colour facia, cost 25 + 5 p&p and can be obtained from Hobbykit Ltd, Unit 19, Capital Industrial Park, Capital Way, London, NW9 0EQ. Tel:(081) 2057485

The second item is a really rapid response from Cliff Jackson to John Foster's query in last month's issue of ***FORMAT*** about where to obtain a Y connector to allow additional peripherals to be fitted to the +2. Cliff says that GREENWELD ELECTRONICS LTD., are offering a skeleton version of the old CURRAH SLOT for the bargain price of £1 + p&p in their latest Bargain List Nº102. Greenweld can be contacted on 0703 236363 and their address is 27 Park Road, Freemantle, Southampton.

Thanks to Gary and Cliff for that information, I'm sure that several readers will be pleased hear of these suppliers.

Well that's all for this month. I realise that the evenings are rapidly drawing out now and that many of you will have less time to spend at the keyboard, but we will still be here to try an answer your questions and pass on any answers you may have, so please send your questions/answers to the following addresses:-

Anything SAM or General Purpose (and, for now, anything Spectrum):-
Ray Bray (Format Help Page),
'Elmsleigh',
4, Tidworth Road,
Porton,
Salisbury,
Wiltshire, SP4 0NG.

Anything +3, CP/M:-
Mike Atkins (Format Help Page),
70, Rudgwick Drive,
Bury,
Lancashire, BL8 1YE.

SMALL ADS

URGENTLY WANTED: The ZXclub Budapest needs the LASER GENIUS by Oasis Software for 128K Spectrum. Phone or Fax to Istvan Ordog 36 1 25 25 025 or write to: H 1148 Budapest, Vezér ut 143, Hungary. We would pay full price.

MESSENGER for sale. Hardly used, £20. Ring 0732 780508 (preferably evenings) or write to Mrs E.Jerrard, 27 Roman Court, Fairfield Road, Borough Green, Kent, TN15 8NY.

SPECTRUM GAMES (including +3 disc games) and magazines. Send SAE for list to A.P.Cresswell, 18 West Steet, Geddington, Northants, NN14 1BD.

WANTED Cable to connect 8056 serial printer to QL or Spectrum+2. Please phone Joe on 0449 675193.

BARGAINS GALORE! Original Spectrum tape software for sale. Prices start from as little as 50p. Over 100 titles available. For full list send SAE to K.H.Tang, 194 Central Road, Coalville, Leicester, LE67 2FG.

FOR SALE: SAM 512K, 2 disc drives (1 external), Sambus, Mouse, Citizen 120D printer, lots of software including Driver, Gamesmaster, MasterDOS/Basic, SC_Assembler, some games, all manuals, £300 o.n.o. Tel: 0750 20175.

BROTHER HR5 printer with mains adaptor and two ribbons. £25. Phone David Field on 0452 411237.

WANTED. Multiface 1 or 128, Videoface and instructions complete for original Microdrive kit. Must be cheap and F.W.O. Phone David on 0538 702723 (7 to 9pm except Mondays).

WANTED Spectrum+2 Machine Language For The Absolute Beginner. (Melbourne) Pay £15 + carriage costs. Phone David on 0656 860550.

YOUR ADVERTS
Selling, Buying, Pen Friends, etc.

Any PRIVATE advert, subject to acceptance, will be printed *FREE* in the next available issue. Any software sold MUST be original. The publishers will not be held, in any way, responsible for the adverts in this column.

Due to shortage of space, if your advert remains unpublished after two months then please send it in again. Trade advertisers contact the publisher for rates.

YOUR LETTERS

Dear Editor,

In the last paragraph of Dr E.J. Wilson's letter in ***FORMAT*** Mar '94, he asks "is there a better way?" (of using his trick for printing £): yes there is; use the same trick as I used for printing a crossed-out equals in my Jan 94 article on redefining the keyboard in *The Secretary*.

He does not say which 'orator' he changes; I assume it is the Lower one, whose 'k-code' (see my article) is 254 (orator caps is 253); so we want to assign this to the key [symbol]L, whose map-position is 196 (SAM User Guide, Appendix, last page). This is done by changing the file "Key_Dkeys" as in the first piece of program in my article with the 2nd line:- `POKE 22752+196,254`

Using the normal key for £ will then send to the printer the code entered in 'orator lower'.

It is also preferable to have £ appearing on the screen, not the symbol indicating 'orator lower': so we change file "Sec_chr2" as described in my article, the essential statement being :-

```
POKE 38648+8*254,12,18,16,56,16,
     16,62,0
```

The suggested code to be entered in 'orator lower' will produce a £ sign, as required, but will also, I think, have the effect that, once a £ is so produced, key [SHIFT]3 will produce £ on the printer, not #; so the USA International Character Set should be restored, making "27,82,3,35,27,82,0" as the code to be put in 'orator lower'; can be done using *The Secretary's* facilities, but see my article for a less laborious way.

Yours sincerely, Ettrick Thomson.

Thank you for the excellent work on *The Secretary* Ettrick, I'm sure many will profit from your efforts. **Bob.**

Dear Editor,

SAM Coupé Public Domain Software Library. At long last, a new Public Domain software library for the SAM Coupé. We aim to provide a reliable PD software supply for all SAM owners. There's no need to call us the 'SCPDSL' or anything like that, just plain and simple SAM PD.

There are NO subscription or membership charges, no clubs or any obligations just, what we hope to be, a reliable software service for SAM users.

We do, however, and always will need new titles for our Library and we will gratefully accept all contributions. Single programs, demos, compilation discs, disc magazines, music modules, Basic, machine code, screen$, in fact we welcome any software, be it PD or Shareware.

From a contributors point of view, donating your work to a PD Library can be a very rewarding experience. You can gain recognition and receive all the encouragement that you need to further your programming experiments. We aren't expecting professional standards

(though chance might be a fine thing!), but we will look at ANYTHING that you send. We will return your disc(s) on request.

To kick-start our PD Library, Steve Taylor, the author of that superb WIMP environment program for SAM - DRiVER, and those excellent machine code articles in FRED magazine, has given us THE MOUSE DRIVER V2.0 - Only £1.50 from SAM PD!

MOUSE DRIVER 2.0, is a utility that we can recommend to each and every SAM user. It is a stand alone version of the mouse pointer routine, as used in DRiVER. All the user need do to use it in his own programs is load in the code, POKE a System Variable and any one of SEVEN different pointers are at your disposal (EIGHT if you include the user definable pointer option!).

XMOUSE, YMOUSE, BUTTON(s) 1 and 3 are all supported. This is, perhaps, the most important PD utility to hit the SAM scene since the (SAMCO) mouse was made available. The program code sits in a Utility page, and like the DUMP utility, MasterDOS/BASIC etc. it will survive NEW, LOAD etc. and is fully interrupt driven. The program is fully documented in a READ ME file on the disc and we can supply printouts on request. Many thanks to Steve Taylor.

Shareware is a concept that is popular on other machines and we would like to encourage this practice for the SAM. The main idea is to help programmer's who may be developing utility software, or even games and also to help people who would like to 'try before you buy'. To meet some of their development costs, time spent programming etc. a programmer can place a freely distributable and unregistered version of their software into the public domain. The idea is that if the user finds that he likes a particular piece of shareware, and if he thinks that the software will be useful, then he can register his interest and pay the programmer a fee (one that is usually requested within the documentation for the program) and thus the programmer will receive a reward for his work. The benefits for the user are that: 1, your conscience will be clear if you like and intend to use the program, and 2, the programmer may send you an enhanced version or updates of the work.

Some programmers release shareware versions of programs, with certain features disabled. This is a good way of 'advertising' the software in the form of a partly working demo. On payment of the shareware fee, the user will then receive a fully working version.

We at Sam PD are also catering for overseas Sam users. If you find that because of the low price of PD software and the added cost of a Euro cheque, it works out expensive to obtain PD software. The next time you order ***FORMAT*** or any other Sam product for that matter, be it hardware or software, then send the whole cheque to us at Sam PD. We will then split your cheque and send you your PD software and pass your cheque and order onto the other distributor for their attention.

For the full SAM PD list, please send a large stamped self addressed envelope to:- Derek Morgan, 18 Mill Lane, Glenburn Road, Old Skelmersdale, Lancashire, WN8 8RH

Yours sincerely, Derek Morgan.

Dear Editor,

An Adventure Probe reader has drawn my attention to the fact that a few people may not be aware that 'The Guild Adventure Software', as run by Tony Collins, has ceased trading, and are therefore still including The Guild in their list of contacts.

I informed my readers that The Guild had ceased trading in the August 1993 magazine (distributed 11th September 1993) and that the Commodore catalogue had already been passed to Binary Zone PD (to the surprise of the authors of the adventures as they hadn't been advised of the change), and that he intended to drop the Spectrum titles.

Since the beginning of September 1993 no one, to my knowledge, has been able to contact Tony Collins. Authors of the adventures he sold haven't received any royalties from him, and many customers haven't received a reply to their long-outstanding orders for software.

More recently some have reported that letters and cheques have been returned by the post office marked "moved house - no forwarding address given". To make matters worse, the glossy magazines, who work 2 to 3 months in advance, continued to run special offers for The Guild software.

I contacted a few authors with the view to publishing some of the titles to ensure thy remained available to players, but soon realized I couldn't offer a full service as I wasn't able to provide the titles on disc. Since then I have been in close association with Phil Reynolds of The Adventure Workshop and it has been agreed that he will publish the titles, as he can offer both disc and tape, while I will only offer (on tape) the adventures by authors who have donated the proceeds to Adventure Probe. All titles have been thoroughly checked, any bugs eliminated, and the presentation improved.

I hope this information will help to clear up some of the confusion that has arisen due to the sudden departure of Tony Collins and The Guild.

Yours sincerely, Barbara Gibb.

Anyone wishing to contact Barbara can do so by writing to 52 Burford Road, Liverpool, L16 6AQ. ***Ed.***

Dear Editor,

May I take this opportunity to say that as a SAM owning, middle aged illiterate, with game playing sons, I find ***FORMAT*** a great help. However, I wish you would review games. I've bought loads of games in a one man attempt to save the SAM and would have welcomed objective reviews. I would also like unbiased guidance on instantly compatible printers re:- price, print quality, running costs etc.,

Yours sincerely, J.Wright.

Games reviews are the one thing we have always kept clear of in ***FORMAT***. This stance came about because when ***FORMAT*** was launched there were 5 or 6 glossy mags all printing reviews of Spectrum games. Now things are different. ***FORMAT*** is the only large circulation printed magazine left with any interest in the Spectrum and SAM market. If I even mention games in these pages some readers go into apoplexy but what do readers in general think, let me know. ***Ed.***

Dear Editor,

I was delighted to see from the current ***FORMAT*** that you are re-marketing the Cardex/PCG DTP suite, although I must say that I think that you have pitched the price a bit high.

Wordmaster, in my view, is infinitely superior to *Tasword* and its derivatives, and I made several desultory attempts to contact PCG to try to persuade them to

put it back on the market. You have obviously been more successful, or more persistent, than I.

I am spending more and more time with my PC these days, and I am seriously considering buying the Spectrum version of *Wordmaster* (I have only got the SAM version) to run under Lunter's emulator (but only in 48K mode!!!)

As my computing horizons have expanded over the years, I have acquired better, or at least newer, hardware and software. But because of my inherently squirrel-like nature, I have been unable to throw the earlier things away, in case I ever needed to come back to them.

But I have now reached an age when, if I am honest with myself, I can virtually guarantee that I will never have the time or the interest to return to the ZX81, and having recently acquired a PC-compatible I suspect that the time I devote to the Spectrum will, before long, reduce drastically.

My problem is what to do with this ageing hardware and software. I feel the need to dispose of it, since I could use the cupboard and bookshelf space that it occupies for other things, but I am reluctant to throw anything like this away until I am certain that it is of no interest to anyone.

Now clearly I could advertise all the items for sale via MicroMart, or the small ads column in ***FORMAT***, but since I am not really looking to convert the items into cash this may not be the best way to go.

Inspired to some extent by recent comments in PCW and Computer Shopper, I wondered whether there was any way in which you, and/or ***FORMAT***, could set up a pool of old Sinclair and SAM hardware and software that could be sold to interested parties for reasonable sums, the proceeds being contributed to a suitable charity or group of charites - obviously if ***FORMAT*** staff were involved, then suitable handling charges would be appropriate.

If there was any mileage in this idea, I could contribute original copies of *Tasword 2* and *Outlet v2.0* for SAM, (both with manuals), a cassette version of *The Last Word* plus manual for a 48k Spectrum, copies of the Spectrum introductory, BASIC and IF1/Microdrive manuals, a VTX5000 modem and two rubber-key 48k Spectrums. If you thought it worthwhile reaching back into history, I also have one and a half ZX81s plus documentation and some programs. In addition there are many Spectrum books sitting on my bookshelves.

What do you, and your readers, think about this idea - and would they support it themselves? If not, then any other suggestion would be welcome - one thought being to donate to a suitable childrens home, which would need to have access to a Spectrum expert to get things up and running.

I have recently dropped my SAM power pack on the floor and broken the top cover - are there any spare covers available.

Yours sincerely, Alan D.Cox.

We have handled charity items before but it is a real bind to try to do anything by mail-order. However, as we have the Gloucester Show at the end of April (with another later in the year I hope) we could set up a charity stand if we got enough donations. Anyone interested should send items, clearly marked as being for charity sale, and clearly marked working or not working. I don't know if there is time for the April show

but let's see what comes in.

SAM power supply cases are still available - see some of the FRS pages from a few months back.

As to the PCG DTP pack being over priced. With a saving of over £23 I thought it was excellent value for money. ***Ed.***

Dear Editor,

First of all, I would like to say that I look forward to receiving ***FORMAT*** each month. Looking back through the back copies as I do while waiting for the latest copy, I noticed a series of articles that I forgot to mention in your questionnaire. The series 'Hardware Design Course' I enjoyed particularly, as it encourages one to explore the possibilities of the SAM even further. I would just like to ask: have we seen the last of these articles, or do you intend to publish further parts to this series? If not, can I obtain these articles anywhere else? Also, in part one the author talks about 'a series of inexpensive kits will also be available as the series progresses'. Are any of these kits still available, indeed is it possible to obtain the Hardware Development Kit that was released prior to Samco's demise? If you could help me with any information I would be most grateful. Another question (sorry): Do you know of any of the following:- 1) When will the Video Digitizer be ready for release. 2) The address of a PD library for the SAM? 3) Was the rumour of a program to load AMIGA screens into the SAM just a rumour, or does such a program exist, and from whom? 4) Has my renewal reached you? because as at time of writing, I haven't received the February copy.

Many thanks for your attention.

Yours sincerely, I.W.Canfield.

Adrian Parker is having trouble finding the time to write the hardware article but he has promised me a couple of months worth very soon. See last months editorial about the hardware kits.

1) Sometime, but don't hold your breath. 2) See earlier in letters section. 3) There are a few routines floating around but nothing commercial. ***Ed.***

And 4) Yes, it was in our 'hold-back' file because it didn't have a membership number on, but now I have it because its at the top of your letter! ***Jenny***.

Dear Editor,

Enclosed is a cheque because I think my subscriptions are about due. How can you tell? Fred Mag has a number on the envelope with your membership number.

I wonder if you could help me, is there any software on the market that could emulate the 128K Spectrum and the Amiga computers. As a workmate with his emulator on his Amiga will emulate some of my SAM Discs.

Thank you for any info and keep up the good work and keep the Spectrum alive.

Yours sincerely B.Tillotson.

We have covered your first question many times in the past but, at the risk of boring some readers, I'll tell you again. The top line of the label used to send out ***FORMAT*** each month contains the following: 5 digit membership number, space, 4 digit expiry date, space, our sort code. The expiry date is the month your membership runs out so, for example, if the code was 0794 this would mean that you expire at the end of July this year and therefore the August 1994 issue would be the last you get. But renewal notices are sent out with the last two issues so there is no danger of you

missing issues.

As to your second question, I must admit I don't quite understand what you are saying. Perhaps you could write again putting the question a little clearer. ***Ed.***

Dear Editor,

In response to Mark Sturdy's letter, I wonder if he and Geoff Winkless are aware of the hardware add-ons that David Ledbury is hoping to bring out on the SAM. The first of these should be the SAM VOX II (SAM VOX I didn't work!), which would give the Coupé a sampler chip and 4 analogue-to-digital channels to enable it to play instrument voices. He also wants to use the proceeds to create a graphics enhancement board, which would give full screen scrolling and hardware sprites (whatever they are), a 20 MHz accelerator chip and a hard drive.

I'm not sure what stage of development these things are at, but I have written to Dave telling him that I want to see them appear on the SAM, and I'd urge all ***FORMAT*** readers to do the same. He may not persevere unless he knows he has the support of the people.

I got all this information from a mailshot that was sent out to all SAM Coupé Adventure Club members. I wanted to pass it on to a wider readership, though I don't know if David has contacted you himself.

Yours sincerely, Nick Xylas.

Now I don't want to be a pessimist Nick, but I doubt very much that these products will ever come off. A hard drive is possible, it has been done on the Spectrum before, but it would require a DOS and that would take a long time and much money to produce.

An accelerator chip or new graphics board are again possible, but the market would be small and therefore the cost very high.

Any readers like to join in the debate? ***Ed.***

Dear Editor,

I have for some time wondered whether to ditch SAM in favour of an MS-DOS PC, due to the lack of software etc., but recent improvements have persuaded me to stick with SAM.

I therefore recently bought a second-hand twin-drive SAM (advertised in ***FORMAT***) as back-up to my original SAM and placed several orders for the mouse, utilities and other software like Lemmings, Legend of Eshan and Sampaint.

I have bought the games mainly on the strength of Bob's endorsement, knowing his normal lack of interest in these things and Sampaint on Carol's recommendation.

Anyway, knowing how important the membership number is to you and the problems it causes when it is not provided or is wrong, it really is ironic that the order to you was the only one with the wrong number.

The correct number is given above and I apologize for the error.

Thanks for the March issue of ***FORMAT*** and for all your hard graft on our behalf.

I can now start to explore Lemmings and Sampaint using my new mouse and look forward to receiving the rest of my order.

Yours sincerely, Maurice J.Smith.

Letters may be shortened or edited to fit on these pages.

Please keep your letters short so we can fit in as many as possible each month.

SOFTWARE

SAM Coupé Software On 3½" Disc

PCG's DTP PACK. The famous package including WordMaster, TypeLiner, HeadLiner. Complete on disc for SAM. **R.R.P. £37.50.** *SPECIAL READER'S BONUS* - Now includes 2 extra font packs worth £15.90 bringing the total pack value to **£53.40.**

Order Code: SCD-04 **Price:** £34.95

COLOUR DUMP. Screen dump software to work with Epson compatible colour printers. RRP £10.

Order Code: SCD-05 **Price:** £6.95

SAMSPEC SOFTWARE

SAM + Spectrum versions on one disc
(80 track 3½ or 5¼ - please state on order)

FONT LIBRARY. 100 screen fonts plus support software, easy to use on either computer.Special +3 disc available (Spectrum files only). RRP £8.95

Order Code: SSD-01 **Price:** £6.95

MONEY MANAGER. Advanced personal budgeting system for both machines. Spectrum version needs 128K/+2 (not +2a or +3). RRP £15.95

Order Code: SSD-02 **Price:** £12.95

SPECTRUM SOFTWARE

Unless otherwise indicated these are DISCiPLE/ PLUS D programs. Please state 3½" or 5¼" disc (80 track only) on order.

ARTIST 2. The best art package ever written for the Spectrum. 48K and 128K versions on the same disc. With page-maker and utility software. Full manual. RRP £19.95

Order Code: SPD-01 **Price:** £16.95

UTILITY DISC #1. Six of the best and most useful disc utilities for DISCiPLE or PLUS D. Added bonus program - not to be missed. RRP £9.95

Order Code: SPD-02 **Price:** £6.95

PCG's DTP PACK. The famous package including WordMaster, TypeLiner, HeadLiner. **R.R.P. £37.50.** *SPECIAL READER'S BONUS* - Now includes 2 extra font packs worth £15.90 bringing the total pack value to **£53.40.**

Order Code: SPD-04 **Price:** £34.95
Special 3" disc version for +3
Order Code: SPD-04C **Price:** £36.95

HARDWARE

SPI -SAM's Printer Interface. A full feature parallel (Centronics) printer interface. Totally compatible with SAM's ROM and DOS printing routines and with utility programs like FLASH, Word-processors, etc. Uses standard BBC type lead (see below). RRP £24.95

Order Code: SPI **Price:** £19.95

>>>>

FORMAT READERS SERVICES

PRINTER LEAD. Top quality parallel printer lead (BBC standard). 2 meters long. For use with DISCiPLE, PLUS D or SAM. RRP £10.95

Order Code: PPL-01 **Price:** £7.95

SAM COUPÉ TECHNICAL MANUAL Version 3. The real inside info on SAM. As published by MGT/SAMCO. No SAM is complete without one. Now only available through us. RRP £16.95

Order Code: STM-01 **Price:** £12.95

MOUSE MAT & HOLDER. Top quality mouse mat (not to be confused with the small cheap ones) together with a handy mouse holder to keep your mouse out of harms way when not in use. Holder fits to any smooth surface with sticky pads provided. RRP £8.98

Order Code: SMM-01 **Price:** £5.95

FORMAT BACK-ISSUES

Volumes 1,2,3 and 4 of FORMAT are now available only as complete volumes.

Order Code: FBI-V1 (Vol.1) **Price:** £10.00
Order Code: FBI-V2 (Vol.2) **Price:** £12.00
Order Code: FBI-V3 (Vol.3) **Price:** £12.00
Order Code: FBI-V4 (Vol.4) **Price:** £12.00

All issues, from Volume 5 Nº 1 to the latest issue, are still available separately. To order just quote the Volume/Issue Nº. **Price:** £1.50 each, (5 or more £1.25 per copy).

FORMAT BINDERS

Hardcover binders to protect your valuable magazines. Each will hold 12 copies of *FORMAT* (or any other A5 magazine or manual). Copies held in place by wires that just spring out for easy insertion or removal.

Order Code: FMB-01 **Price:** £4.50

ORDERING All prices include UK p&p, overseas readers please add 10% to total order for extra postage.

Clearly state Order Code, Product description, Quantity required and Price. Remember to add any postage and don't forget your Membership Number or we can't process your order. Payment in STERLING by Cheque (drawn on a UK bank), P.O. Euro Cheque or Cash. Make cheques payable to FORMAT. Payment MUST be in same envelope as order. **Send to FORMAT's address on page 3.** Normally dispatch is at the same time as your next issue of FORMAT. We will not be held liable for delays or non-delivery due to circumstances beyond our control.